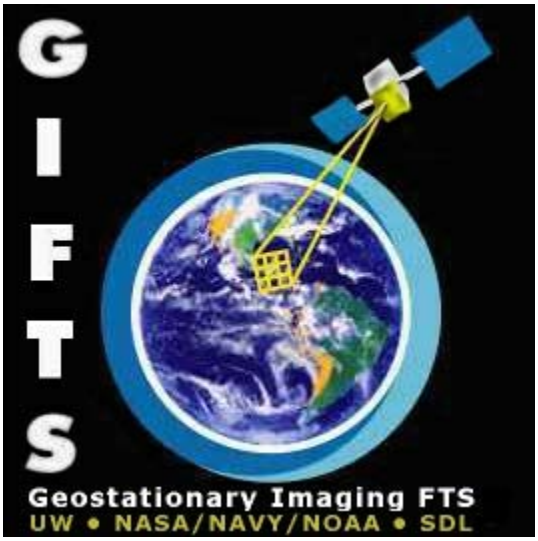

Geosynchronous Imaging Fourier Transform Spectrometer (GIFTS) – Indian Ocean METOC Imager (IOMI) Mission

Software Project Management Plan

Signature Page

Prepared by:

Signature on file

John C. Hinkle Date

GIFTS-IOMI Software Manager/Architect

Concurred by:

Signature on file

John Battcher Date

GIFTS-IOMI System Manager

Signature on file

Leslie Johnson Date

GIFTS-IOMI Software QA

Approved by:

Signature on file

F. Wallace Harrison, LaRC

GIFTS Project Manager

Document Revision History

Revision	Date	Description
1.0	2/28/01	Initial Release
1.1	11/6/01	Minor modifications to reflect changing project attributes (approved 5/28/2002)

TABLE OF CONTENTS

1 PURPOSE

1.1 DOCUMENT IDENTIFICATION

The Geosynchronous Imaging Fourier Transform Spectrometer (GIFTS) Software Project Management Plan (SPMP) identifies, establishes, and governs the management approach to be applied to all software developed for the GIFTS–Indian Ocean METOC Imager (IOMI) project. The SPMP tailors generic software project management and product engineering guidelines to fit the needs of the GIFTS-IOMI project. This plan identifies policies, procedures, and guidelines used to facilitate an effective software engineering effort throughout the software development lifecycle. The SPMP includes planning information for the conduct of software independent verification and validation (IV&V) for the GIFTS-IOMI project.

1.2 DOCUMENT GOVERNANCE

The Langley Management System LMS-CP 5528 Software Planning, Development Acquisition, Maintenance, and Operations (ISO 9001 certified) provides the structure for NASA Langley software planning. The SPMP is written in accordance with these guidelines. According to those guidelines, the GIFTS-IOMI project software is of "High" criticality.

The SPMP applies to software developed by the National Aeronautics and Space Administration (NASA) Langley Research Center (LaRC) and all GIFTS-IOMI contractors intended for delivery as a flight system, ground system, data product generation, or test equipment, or to be used in conjunction with integration or formal qualification testing. The management processes described in the SPMP will be implemented after successful completion to the GIFTS-IOMI Preliminary Design Review.

1.3 PROJECT OVERVIEW

The GIFTS-IOMI Mission is a team effort being led by NASA LaRC. Co-Investigators reside at the Utah State University's Space Dynamics Laboratory (SDL) and the University of Wisconsin (UW). LaRC has overall mission management responsibility, including final selection of the spacecraft and launch vehicle providers and technologies to be incorporated into GIFTS-IOMI, and final instrument integration and environmental testing. The GIFTS-IOMI Instrument consists of the Sensor Module (SM) and the Control Module. LaRC is responsible for the design, development, integration, and test of the Control Module. SDL will design, develop, and fabricate the SM and associated Ground Support Equipment (GSE), perform instrument calibration, and support integration and test activities. UW will provide algorithm development, meteorological data analysis, and educational/public outreach associated with the mission.

1.4 EVOLUTION OF THE SPMP

The SPMP will be baselined prior to the GIFTS-IOMI Preliminary Design Review (PDR) and will be maintained under Configuration Control in accordance with the GIFTS-IOMI Configuration Management Plan (CMP). The SPMP will be reviewed for potential revision prior to each major GIFTS-IOMI project milestone, and upon major change in the project that affects the software management approach.

1.5 DEFINITIONS

- Computer Software Configuration Item (CSCI)—an aggregation of software that is designated for configuration management and treated as a single entity in the configuration management process.
- Computer Software Component (CSC)—a functionally or logically distinct part of a CSCI; typically an aggregate of two or more software units.

1.6 DOCUMENT ORGANIZATION

This SPMP is organized as follows:

- Section 1 identifies the SPMP, discusses its scope, and describes briefly the GIFTS-IOMI project

- Section 2 identifies the documents referenced in the SPMP.
- Section 3 provides an overview of the software, organization, and roles and responsibilities within the GIFTS-IOMI project.
- Section 4 defines the plans for managing software across all of the GIFTS-IOMI project.
- Appendix A provides the list of acronyms used in this document.
- Appendix B details the metrics to be collected for each software development.

2 REFERENCE MATERIALS

2.1 GIFTS-IOMI PROJECT DOCUMENTATION

a. GIFTS-IOMI Continuous Risk Management Plan, document 01-005, latest revision

b. GIFTS-IOMI Mission Assurance Plan, 01-006, latest revision

c. GIFTS-IOMI Software Quality Assurance (SQA) Management Plan, 01-010, latest revision.

d. GIFTS-IOMI Software Configuration Management Plan, document 01-013, latest revision

e. GIFTS-IOMI Software Integration, Test, and Verification Plan, document 03-018, latest revision

f. Software Review Procedures for the GIFTS-IOMI Project, document 03-xx, latest revision.

2.2 OTHER GIFTS-IOMI DOCUMENTATION

a. GIFTS-IOMI Project/Software IV&V Facility Memorandum of Agreement (MOA), TBD

2.3 NASA DOCUMENTATION

a. Recommended Approach to Software Development, Revision 3, June 1992, Software Engineering Laboratory Series, SEL-81-305.

b. Software Planning, Development, Acquisition, Maintenance, and Operations, LMS-CP-5528 Revision B

c. Software Acquisition Planning, LMS-CP-5532, Revision B

2.4 INDUSTRY DOCUMENTATION

a. Institute of Electrical and Electronics Engineers (IEEE) Standard 1058-1998, *IEEE Standard for Software Project Management Plans*.

b. IEEE/Electronic Industries Association (EIA) 12207.0-1996, IEEE/EIA Standard, Industry Implementation of International Standard International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC) 12207: 1995, *Standard for Information Technology – Software Life Cycle Processes*.

c. IEEE/EIA 12207.1-1997, IEEE/EIA Standard, Industry Implementation of International Standard ISO/IEC 12207: 1995, *Standard for Information Technology – Software Life Cycle Processes – Life Cycle Data*.

d. Software Engineering Institute at Carnegie Mellon University, *Continuous Risk Management Guidebook,* 1996, NTIS#: AD-A319533KKG, DTIC#: AD-A319 533\6\XAB.

3 SOFTWARE OVERVIEW

3.1 SCOPE

All software acquired or developed by the GIFTS-IOMI project are within the scope of this SPMP. This includes GIFTS-IOMI instrument, operations, ground data processing, support, integration, acceptance, and maintenance software and associated documentation.

3.2 ORGANIZATION AND RESPONSIBILITY

The GIFTS-IOMI Software is managed by the GIFTS-IOMI Software Manager who reports to the GIFTS-IOMI Systems Engineering Manager. Overall GIFTS-IOMI software system development planning, organizing, monitoring, controlling, and reporting are the primary responsibilities of GIFTS-IOMI Project Office Software Engineering . The Software Engineering personnel ensure that all software development meets requirements, is delivered on time, and within budget. Software Quality Assurance (SQA) is performed in accordance with the GIFTS-IOMI SQA Management Plan. Software Independent Verification and Validation (IV&V) is performed by the NASA Goddard Space Flight Center (GSFC) Software IV&V Facility in accordance with the GIFTS-IOMI Project/Software IV&V Facility Memorandum of Agreement (MOA). The makeup and staffing of each software team will be identified in the Software Development Plan of each software provider where, also, the scope of the software effort is quantified.

Table 1 shows the software currently within the scope of the SPMP, as well as the providers of same.

Table 1 Software and Providers

Software	Provider
Instrument Controller Flight Software	**Alaskan Slope Regional Corp (ASRC)**
Digital Signal Processing Flight Software	**LaRC**
Control Module Flight Software Simulator	**ASRC**
Sensor Module Software	**SDL**
Sensor Module Software Simulator	**SDL**
Ground Systems Software	**Zel Technologies**
Ground data processing software	**University of Wisconsin/LaRC**
Ground Support Equipment	**LaRC**

3.3 SCHEDULE

The official GIFTS-IOMI Integrated Software schedule will be developed and maintained by the GIFTS-IOMI Project Office Software Engineering team, who will ensure it is consistent with the GIFTS-IOMI Integrated Master Schedule that is under the control of the GIFTS-IOMI Engineering Review Board. The Integrated Software schedule will be

configuration controlled in accordance with the GIFTS-IOMI CMP. The Integrated Software schedule will be reviewed monthly with all software providers, as well as with project engineering. The schedule will reflect the following information:

- Major software developmental milestones for each CSCI
- All CSCI integration and test activities
- All CSCI major software reviews
- All software schedule dependencies
- All major GIFTS-IOMI reviews
- All major GIFTS-IOMI milestones
- High level milestones and events of hardware dependencies

4 COMMON SOFTWARE MANAGEMENT REQUIREMENTS

Given the number of geographically dispersed and dependent software developments, it is important that there be rigor, flexibility, and consistency in their development and management. This section describes the approaches defined by the GIFTS-IOMI project for the comprehensive management of the diverse GIFTS-IOMI software developments. Developers will find assistance in planning their development documented in NASA's *Recommended Approach to Software Development.* This assistance includes guidance for document and review contents.

4.1 SOFTWARE DEVELOPMENT PLANNING

All software developments funded by and delivered to the GIFTS-IOMI Project will be conducted in accordance with a Software Development Plan (SDP) written by the software developing organization and approved by the GIFTS-IOMI Project. The SDP's will be documented following the guidance provided in IEEE Standard 1058.1-1997, *IEEE Standard for Software Project Management Plans*. The SDP will explicitly identify the CSCI's to be provided by the developer and the life-cycle models employed for development.

4.2 SOFTWARE REQUIREMENTS MANAGEMENT

All software requirements will be traceable to parent requirements. This traceability will be documented in the software requirements specifications. Once software requirements are put under configuration management, they will be maintained utilizing a GIFTS requirements management tool (e.g., DOORS). This tool will be utilized for traceability, configuration management, and verification tracking. The output of the tool will be utilized during the software Functional Configuration Audits.

4.3 MANAGEMENT

4.3.1 Monthly Status Reporting

Each software development organization will produce and deliver a Monthly Software Status report each month. This report will identify the status of the development effort. It will also identify the developer's assessment of project software development progress during the reporting period as measured against planned progress (in accordance with updates to the metrics defined in Appendix B of this SPMP) and milestone achievement, and present the planned project development activities for the next period. The report will identify significant technical activities from the previous month.

Also, this report will identify and status a list of significant developer software risks and the developer's proposed mitigation. Statusing will continue on each item on the list until such item is removed. Rationale for all additions/deletions to the list will be reported.

This report will describe, as well, unanticipated problems encountered by the provider, solutions to problems, expediting techniques/methods used, and actions taken to prevent recurrence of the problems (i.e., lessons-learned). If no reportable lessons have been learned during the reporting period, the monthly report transmittal letter will state that no Lessons-Learned Reports are being forwarded.

Separately, each software development organization will meet (LaRC remote organizations electronically) with the GIFTS-IOMI Project Office Software Engineering team on a monthly basis to discuss the Monthly Software Status report. All materials (including metrics) to be discussed at the monthly meeting should be electronically provided to the GIFTS-IOMI Project Office Software Engineering team three business days prior to the meeting. The meeting will be chaired by a member of the GIFTS-IOMI Project Office Software Engineering team and will address the following:

a. Monthly Status Report.

b. Proposed changes to the Software Integrated Schedule

c. Action items from the previous meetings.

d. Special topics, as necessary.

4.3.2 Progress Tracking

The software development will be monitored to identify critical paths and avoid schedule delays.

4.3.2.1 Schedules and Milestones

The GIFTS-IOMI Software Manager in accordance with the GIFTS-IOMI Project Schedule will monitor development of the GIFTS-IOMI software products. The Software Manager will track the schedule and milestones and provide regular reports on the software progress against the milestones. The schedules, milestones and reports will show interdependencies with other GIFTS-IOMI disciplines. This information will be available for review at the project level for insight into the software development progress.

Each software provider will develop estimates of cost, effort, and schedule for their CSCI. In addition, the GIFTS-IOMI Project will prepare a master schedule, to be used by each provider, for delivery of all products. The GIFTS-IOMI Project will, after negotiation with the provider, agree on a Work Breakdown Structure (WBS) breakdown of cost and on schedules. The GIFTS-IOMI Project will assess progress against the agreed to WBS resource use profiles and schedules by evaluating provider progress reports. At a minimum, each flight and ground operational software CSCI will be a separate WBS element.

4.3.2.2 Progress Metrics

Each developer will provide updated metrics in accordance with Appendix B of this SPMP on a monthly basis. These metrics will be reported in the Monthly Status Reports. To ease reporting, analysis, and integration, the GIFTS-IOMI Project Office Software Engineering team will work with the developers to create a common metrics reporting template and medium. This common template will take into account the fact that not all metrics will be necessary for each development.

4.3.3 Weekly Tag-Ups

The GIFTS-IOMI Software Manager and the leads of each software development organization will have a weekly telecon to ensure coordination and timely resolution of issues.

4.3.4 Control Boards and Working Groups

As required, various software specific boards and working groups will be formed with affected members of the GIFTS-IOMI software and systems community, as well as interfacing organizations. Examples include Software Interface Control Working Groups (ICWG's) and Software Control Boards. It is the initial intent of the GIFTS-IOMI Project Office Software Engineering team to utilize existing project boards and working groups and only form software specific boards and working groups where the volume of effort dictates a separate software focus. To this end, a representative of the GIFTS-IOMI Project Office Software Engineering team will be a member of the following project teams:

- GIFTS-IOMI Engineering Review Board
- Systems Engineering Working Group
- Payload Accommodations Working Group
- Mission Operations and Ground Systems Working Group
- Test and Verification Working Group

4.3.5 Management Documentation

The GIFTS-IOMI Project Office Software Engineering team is responsible for the development and maintenance of a number of documents for the project relating to software in addition to this SPMP:

- Software Integration, Test, and Verification Plan
- Instrument Controller Software Requirements Specification
- Digital Signal Processing Flight Software Requirements Specification
- Software GIFTS to IOMI Transition Plan

4.4 DOCUMENTATION REQUIREMENTS

Each developing organization will document each CSCI through the development and provision of documentation which address the following (note: tailoring and explicit mapping of this list to software deliverables will be documented in the individual SDP's):

- Software Development Planning
- Software Requirements
- Software Development Standards
- Software Architecture
- Software Design
- Software Code
- Software Unit Test Plans, Procedures, and Reports
- Software Integration Plans, Procedures, and Reports
- Software Acceptance Plans, Procedures, and Reports
- Software Interface Definition
- Simulation/Support Software Requirements
- Simulation/Support Software Code
- Simulation/Support Software Verification/Certification Plans, Procedures, and Reports
- User/Operator Procedures
- Maintenance Planning and Procedures
- Software Configuration Management and Control
- Software Quality Assurance
- Version Description

4.5 REVIEWS/AUDITS/INSPECTIONS

4.5.1 Peer Reviews

Each of the above documents will undergo a working level, peer review that involves, at least, representatives from the developing organization, and the GIFTS-IOMI Project Office Software Engineering team. Invitees will include representatives from organizations representing GIFTS-IOMI interfacing elements to the CSCI, Software Quality Assurance, and GIFTS-IOMI Software IV&V personnel as well as the GIFTS-IOMI element within which the CSCI operates. The peer review will be held prior to any formal review (e.g., Software Requirements Review (SRR)) of the document.

4.5.2 Formal Reviews

Formal software reviews will occur throughout the GIFTS-IOMI Software development cycle, from initial requirement analysis through final software acceptance. All formal reviews share the common traits of agreeing upon detailed review contents and pass/fail criteria before the review, electronically distributing material at least ten working days prior to the review, assigning action items during the review, and follow-up regarding the results of the review. Resolution and subsequent closure of all action items is required before the review is considered successfully completed. Formal reviews will be chaired by a representative of the GIFTS-IOMI project who will determine whether the review has passed, provisionally passed (i.e., will be considered passed after certain actions are completed), or failed. If a review has failed, a delta review is required. At the discretion of the chair of the review, a delta review may be required if the review received

a status of provisionally passed. The procedures by which all formal software review are to be conducted are documented in the Software Review Procedures for the GIFTS-IOMI Project.

In addition to the following software reviews, developers are expected to support GIFTS-IOMI system reviews through both presentations and participation to ensure that software is successfully integrated into project efforts. Participation in appropriate reviews with the spacecraft developer will also be required.

4.5.2.1 Software Requirements Review

The Software Requirements Review (SRR) is the forum for presenting, discussing and agreeing upon the software requirements derived from GIFTS-IOMI system requirements and documented in the Software Requirements document. During the review, an overview of the software and operational environment will be provided, along with discussions of software functionality that result from satisfying the requirements. Other discussion topics include system and software requirement traceability, key performance requirements, initial timing and sizing estimates, external interfaces and planning for testing the software.

The Software Requirements Review serves two purposes: 1) it ensures that the documented requirements are complete, accurate and in agreement with the system requirements, 2) it ensures that the location(s) and method for testing each software requirement is specified. Successful completion of the SRR (and closure of corresponding action items) signifies approval of the software requirement and agreement to continue preliminary design.

A GIFTS-IOMI project goal is to hold a single flight software SRR and a single ground software SRR.

4.5.2.2 Preliminary Design Review

The purpose of the GIFTS-IOMI Software Preliminary Design Review (PDR) is to describe the early design of the GIFTS-IOMI software that was developed from the approved set of requirements. The task structure, external hardware and operational interfaces, and operating system usage will be reviewed. Other discussion topics include the flowdown of CSCI-level requirements to the software component-level and formal qualification tests, refined timing and sizing estimates, and test plans. Determining if the preliminary design fully satisfies the software requirements is a major focus of PDR. Successful completion of the PDR (and closure of corresponding action items) signifies approval of the early flight software design and permission to continue detailed design.

A GIFTS-IOMI project goal is to hold a single flight software PDR and a single ground software PDR.

4.5.2.3 Critical Design Review

A software Critical Design Review serves three purposes: 1) it ensures that the proposed software detailed design satisfies all of the CSCI and CSC requirements, 2) the detailed design is examined to ensure high software design quality, and 3) it provides an opportunity for necessary discipline interaction within the GIFTS-IOMI project, independent peer review, visibility into the software design, education of the software team and maintenance of commonality. Successful completion of the CDR (and closure of corresponding action items) signifies approval of the detailed design and permission to proceed with the code and unit test phase.

4.5.2.4 CSCI Integration Test Readiness Review

A CSCI Integration Test Readiness Review (TRR) is conducted by executing the CSCI Integration Plan and Procedures prior to turning over the CSCI to LaRC for integration and test. The primary objectives of the review are to ensure all planned external CSCI interfaces have been tested, the CSC's function as an integrated CSCI (or portion thereof), and the software is ready for integration with other software and with hardware. Note that the CSCI Integration TRR is required for flight software only given that the ground software will be integrated by the Ground System developer.

4.5.2.5 Formal Qualification Test TRR

A Formal Qualification Test (FQT) TRR will be conducted prior to the formal run of the FQT that verifies the requirements for each CSCI. The review will ensure that the FQT Plan and Procedures have been verified and approved, the software is under appropriate configuration management control, and all problems have been identified and resolved. Successful completion of the TRR must occur before FQT may begin.

4.5.2.6 Functional Configuration Audit

The Functional Configuration Audit (FCA) is conducted following the successful completion of the test procedures. The goal of the FCA is to ensure that the software performed at a level that validates system operation, verifies requirements traceability and satisfaction, ensures that the tested software baseline is under appropriate configuration management control, and all potential problems have been identified and resolved. Formal acceptance of the product follows successful completion of the FCA.

4.6 SOFTWARE INTEGRATION AND TEST

Software Integration and Test is the responsibility of the GIFTS-IOMI Project. The GIFTS-IOMI Project is responsible for ensuring that all CSCI's operate collectively to meet software project and mission requirements. The GIFTS-IOMI Project will produce integration test plans, procedures and reports in accordance with IEEE 12207 standards to meet all testing requirements as described in the GIFTS-IOMI Mission Integration, Test, and Verification Plan. All test plans and procedures will undergo peer reviews and a formal baselining process. This process will be described in the GIFTS-IOMI Software Integration, Test, and Verification Plan.

4.7 SOFTWARE ACCEPTANCE

GIFTS-IOMI Project acceptance of software will occur following the FQT and FCA of that software. Acceptance testing will be conducted as described in the GIFTS-IOMI Software Integration, Test, and Verification Plan. Objective Pass/Fail criteria will be determined for software acceptance. These criteria will include the permissible numbers of Problem Reports (PR's) of different criticality for the accepted software. More information on acceptance criteria will be included in the GIFTS-IOMI Software Integration, Test, and Verification Plan.

4.8 SOFTWARE STANDARDS, PRACTICES, AND CONVENTIONS

The processes, activities, and tasks associated with each software development will be documented in the individual SDP's. Processes, activities, and tasks followed in implementing the software life cycle should be as described in IEEE/EIA 12207.0-1996, or where the developer has a compatible process in place.

For each CSCI, the documentation approach will be documented in its associated SDP. The areas that must be addressed by documentation are as defined in Section 4.3 of this SPMP, and described in IEEE/EIA 12207.1-1997. Depending on how the life cycle is tailored, not all life cycle data and documentation will be required, and all life cycle data need not be in the form of separate documents (Refer to IEEE/EIA 12207.1 section 4.3, table 1 for the full listing of life cycle data). For consistency of terminology across the GIFTS-IOMI developers, IEEE/EIA 12207.1 terminology will be used where feasible for document names.

4.9 SOFTWARE CONFIGURATION MANAGEMENT

Configuration Management (CM) of all software artifacts (i.e., documentation, operational software, support software) will be the responsibility of the individual developing organizations until delivered to LaRC for baselining (in the case of documentation) or for integration (for software). When LaRC assumes CM of a software artifact, the developer will be responsible for all submissions and updates of the artifact through the LaRC GIFTS-IOMI CM system in accordance with the GIFTS-IOMI CM Plan. The details of the CM system for GIFTS-IOMI software are provided in the GIFTS-IOMI Software CM Plan.

4.10 RISK MANAGEMENT

Each software developer will report and status risks each month in their Monthly Software Status report. Risk description, classification, processing, and closure will be in accordance with the GIFTS-IOMI Risk Management Plan. Note: NASA

has adopted agency-wide a risk management process described in NASA Policy Guidance (NPG) 7120.5A and based upon the Carnegie Mellon University Software Engineering Institute's *Continuous Risk Management Guidebook*.

4.11 SOFTWARE QUALITY ASSURANCE

Overall GIFTS-IOMI project Software Quality Assurance (SQA) will be managed by the LaRC Office of Safety and Mission Assurance. The SQA approach is documented in the GIFTS-IOMI SQA Management Plan.

4.12 SOFTWARE IV&V

Immediately subsequent to GIFTS-IOMI PDR, a Memorandum of Agreement (MOA) will be negotiated with the NASA Software Independent Verification and Validation (IV&V) Facility for the performance of Software IV&V on the GIFTS-IOMI Project. The MOA will document the working relationship, roles and responsibilities, and points of contact necessary to assure mutual benefits to the parties involved. The following subparagraphs describe the associated planning relating to the IV&V performance.

4.12.1 Scope of the IV&V Effort

The scope of the Software IV&V involvement will be documented in the MOA. At this time, IV&V is anticipated on the flight software as well as selected portions of the ground software. IV&V on the LaRC integration, test, and acceptance is also anticipated.

4.12.2 Duration of the MOA

This MOA will be in effect until the completion of the agreed tasking or until terminated at the request of one or both parties. During this time the IV&V Team and GIFTS-IOMI Project Representatives will review the content of the MOA at least once a year for applicability to the GIFTS-IOMI Project and make changes as required.

4.12.3 Schedule and Milestones

The IV&V activities are tied to the development schedules of the GIFTS-IOMI project. The IV&V Team will respond to milestone activities within the GIFTS-IOMI organization through various reports and analyses of the Project. The detailed IV&V activities, reports, and analyses associated with project milestones are described in the GIFTS-IOMI IV&V Project Plan that will initially be provided to the GIFTS-IOMI project by the IV&V Facility 30 days after the MOA goes into effect.

4.12.4 Roles and Responsibilities

4.12.4.1 NASA IV&V Facility

The NASA IV&V Facility personnel will provide the technical direction and financial management for the civil servants and IV&V contractors located at the IV&V Facility, as well as IV&V contractor personnel at GIFTS-IOMI project and developer locations, to perform the tasks listed in this MOA. The NASA IV&V Project Manager (PM) will be responsible for the direction and activities performed by the contractor and for planning and approving the work to be performed, including utilization of IV&V tools available from the IV&V Facility.

The NASA IV&V PM is responsible for assuring transmittal of the IV&V deliverables to the GIFTS-IOMI Project. The NASA IV&V PM will be the formal interface between the IV&V contractor, the GIFTS-IOMI development organizations, and the GIFTS-IOMI Project Office for issues related to the IV&V work being performed for the GIFTS-IOMI project.

4.12.4.2 GIFTS-IOMI Project

The GIFTS-IOMI Project will facilitate the tasks to be performed with the IV&V Facility. This may involve coordination between project personnel and the IV&V facility (as required) to:

- Provide IV&V personnel onsite facilities to perform the IV&V tasks.
- Provide necessary development and supporting documentation to the IV&V personnel to perform the IV&V tasks.

- Transfer the funding to GSFC to fund this activity.
- Provide an IV&V liaison (the GIFTS-IOMI Software Manager) who will act as the project point of contact for formal interactions between the IV&V Team and the project.

4.12.4.3 General Roles and Responsibilities

The following roles and interfaces are defined to ensure that the IV&V Team has adequate access to necessary deliverables and resources and that IV&V results are available to the GIFTS-IOMI Project.

1. GIFTS-IOMI Project organizations and the IV&V Team will interface formally through the NASA IV&V PM for items related to IV&V scope, IV&V product issues, IV&V priorities and schedules, budgets, requirements for access to resources, and delivery of formal IV&V products.

2. The IV&V team members will interface informally through participation in GIFTS-IOMI Team working groups, or their equivalent, as non-voting team members.

    - Working Group participation will ensure that the IV&V Team is provided an interface with the GIFTS-IOMI Team to support informal and formal software development reviews for all life cycle phases.

    - Working Group team leaders will ensure that the IV&V Team has the opportunity to participate in team activities as non-voting team members. Participation in GIFTS-IOMI teams will be accomplished through locally resident IV&V personnel and via teleconferences.

3. The IV&V team will have access to deliverables and resources pertinent to the IV&V tasking:

    - Draft and finalized deliverable and non-deliverable documentation will be made available to the IV&V Team through participation with GIFTS-IOMI Teams responsible for product development. Non-deliverable or informal documentation (i.e. Software Development Folders, Software Interface Requirements Documents, etc.) will be made available at the GIFTS-IOMI facilities. When available, electronic access to GIFTS-IOMI software development documentation and information will be provided as the information is made available to the GIFTS-IOMI teams.

    - Source code documentation will be made available through participation with GIFTS-IOMI teams responsible for product development. When available, electronic access to source code will be provided as the information is made available to the GIFTS-IOMI teams.

    - GIFTS-IOMI Project and developer status reports will be made available to the IV&V Team.

4. The IV&V team will conduct assessments of draft and final GIFTS-IOMI Project documentation, products and processes in coordination with the appropriate level team reviews prior to formal reviews. These reviews, and the IV&V team interactions, will be conducted within the general guidelines and in-process review format established for all GIFTS-IOMI team members. The IV&V Team will ensure that the assessments are at the appropriate level required to support the current development phase and software maturity. During these preliminary assessments, product redlines will be the preferred method of providing IV&V input to the GIFTS-IOMI teams.

5. The IV&V team members will participate in formal project reviews including but not limited to project equivalent reviews of system concepts and requirements, system designs, system software requirements and designs, code, test plans, Test Readiness, Delivery Acceptance, and Operational Readiness for software in the Catastrophic/Critical/High Risk Functions List (CFL). The IV&V team will be given the opportunity to present the results of their analysis at the reviews.

6. Issues, which cannot be resolved at the working group level within an appropriate timeframe, will be elevated to the GIFTS-IOMI Software Manager. The IV&V Team will ensure that the appropriate working level teams are given

sufficient time to respond to IV&V issues, and that the working level teams are also notified that an issue is being raised to the next level. Issues will be elevated when it is clear to the IV&V team that the unresolved issue will adversely affect future technical work or system operations and the issue cannot be resolved within the working group environment. The GIFTS-IOMI IV&V Project Manager will raise issues, which cannot be resolved, to GIFTS-IOMI project management. IV&V representatives and GIFTS-IOMI Project Management will meet as required to discuss issues.

7. IV&V Team requests for additional activities or resources, that present potential impacts to project cost or schedule as determined by the GIFTS-IOMI Project Management or contractors/subcontractors, will be coordinated through the GIFTS-IOMI Project Management and the GIFTS-IOMI IV&V Project Manager.
8. The IV&V Team will make available copies of all technical, issue tracking, and status reports to NASA and the GIFTS-IOMI contractors/subcontractors.
9. The GIFTS-IOMI Project Manager is encouraged to contact the IV&V Facility Director directly should issues need to be discussed that cannot be resolved with the GIFTS-IOMI IV&V Project Manager.

4.12.5 Tasks

The NASA IV&V Facility will perform IV&V on selected Catastrophic/Critical/High Risk (CCHR) GIFTS-IOMI software components. IV&V uses the Catastrophic/Critical/High Risk Functions List (CFL) to define these selected software components and their analysis level. The analysis will consist of requirements, design, code, test, and interface analysis, as applicable for the life cycle of the software being analyzed. In particular, representative tasks for selected software may include:

- Software Functional Requirements Analysis: engineering analysis of functional requirements for completeness, correctness, testability and traceability to system level requirements.
- Software Design Analysis: analysis of the design for completeness, correctness and traceability to functional requirements.
- Software and System Milestone Review Support
- Static code Analysis (by hand and using automated tools): Through the use of automated tools and code language expertise, IV&V will look in such areas as:
    - indicators of software that may be overly complex, difficult to test, or likely to contain latent errors.
    - memory allocation
    - undesirable I/O
    - additional code statements that could cause delays and task blocking
- Unit test analysis: ensure all paths and modules are thoroughly unit tested.
- Software Test Readiness Review Milestone support
- Review of V&V and Test Plan: ensure the V&V and Test plan provides for complete verification and validation of the software, meeting not only the letter of the requirements but the larger functionality issues and including compliance with associated NASA level requirements such as system safety.
- Analysis of software system test procedures: analysis of the completeness of software system testing and traceability to software system level requirements.
- Review of system test results: ensure complete verification of requirements, compliance with test procedures and otherwise ensure the system test met the needs of the project for verification and validation of the software.
- Special Studies as determined by the GIFTS-IOMI project and IV&V Facility

4.12.6 Deliverables

The IV&V Facility will provide the results of the IV&V analyses, identified issues and risks, as well as status reports, to the GIFTS-IOMI Project Management per the following table. (Note: issues and problems are immediately communicated to the appropriate working group, the IV&V Team does not wait for the formal Review and Analysis Reports to begin the issue resolution process).

Product	Recipient	Schedule
GIFTS-IOMI IV&V Project Plan	• GIFTS-IOMI Project Manager • GIFTS-IOMI Software Manager	Draft: 30 days after IV&V initiated. Final: 4 months after MOA. Updates as required, but not less than yearly.
IV&V Technical Issues	• Cognizant GIFTS-IOMI project development team or organization	Ad-hoc, as authored
Monthly Software Status Report • Development Progress Assessment • Operational Constraints • Top Risks • IV&V Contributions	• GIFTS-IOMI Project Manager • GIFTS-IOMI Software Manager	End of Month + 7 working days
Status Briefings	• GIFTS-IOMI Project Manager • GIFTS-IOMI Software Manager	An agreed upon schedule, as desired by the GIFTS-IOMI project.
Catastrophic/Critical/High Risk Functions List (CFL)	• GIFTS-IOMI Project Manager • GIFTS-IOMI Software Manager	4 months after MOA and every 6 months, thereafter
IV&V Performance Briefings • Cost, Schedule, and Performance	• GIFTS-IOMI Project Manager • GIFTS-IOMI Software Manager	An agreed upon schedule, as desired by the GIFTS-IOMI project.

4.12.7 Resources/Budget

4.12.7.1 GIFTS-IOMI Project Resources

The following GIFTS-IOMI Project resources will be made available to the IV&V Team:

1. Access to the GIFTS-IOMI internal web site from Fairmont.
2. Personnel Space on-site at LARC for up to one person.
3. General office materials, telephone and computer electronic access (i.e., e-mail and internet connection) for GIFTS-IOMI on-site IV&V individual(s).
4. Access to GIFTS-IOMI development, test, and implementation facilities for the purpose of reviewing work products and evaluation of testing.

5. Access to development and test computing environments and tools on an as available basis for the purpose of special analyses.
6. Access to problem tracking and Configuration Management Systems.
7. Area access permits for those areas that are accessed by GIFTS-IOMI development and operations personnel.

4.12.7.2 IV&V Resources

Access to the IV&V analysis tools being used on or developed by the IV&V team for this effort will be made available to the GIFTS-IOMI Project on an as available basis.

APPENDIX A: ACRONYMS

ASRC – Alaskan Slope Regional Corporation

CCHR – Catastrophic, Critical, and High Risk

CFL – Critical Functions List

CM – Configuration Management

CMM – Capability Maturity Model

CMP – Configuration Management Plan

COTS – Commercial Off-The-Shelf

CPU – Central Processing Unit

CSC – Computer Software Component

CSCI – Computer Software Configuration Item

EEPROM – Electronically Erasable Programmable Read Only Memory

EIA – Electronic Industries Association

EP – Equivalent Person

FCA – Functional Configuration Audit

FQT – Formal Qualification Test

GIFTS – Geosynchronous Imaging Fourier Transform Spectrometer

GSE – Ground Support Equipment

GSFC – Goddard Space Flight Center

ICWG – Interface Control Working Group

IEC – International Electrotechnical Commission

IEEE – Institute of Electrical and Electronics Engineers

I/O – Input/Output

IOMI – Indian Ocean METOC Imager

ISO – International Organization for Standardization

IV&V – Independent V&V

K – Thousand

LaRC – Langley Research Center

LMS – Langley Management System

MOA – Memorandum of Agreement

NASA – National Aeronautics and Space Administration

PDR – Preliminary Design Review

PM – Project Manager

RAM – Random Access Memory

SCE – Software Capability Evaluation

SDL – Space Dynamics Laboratory

SDP – Software Development Plan

SEI – Software Engineering Institute

SM – Sensor Module

SPMP – Software Project Management Plan

SRR – Software Requirements Review

SQA – Software Quality Assurance

TRR – Test Readiness Review

UW – University of Wisconsin

V&V – Verification and Validation

APPENDIX B: SOFTWARE METRICS

Required NASA Software Metrics

- Software Characteristics (Section 1.0) are captured in the initial metrics report. (Note: If the data requested is not available at initial submission, complete it at the next monthly reporting period.)
- Software Planning, Tracking, and Oversight Data (Section 2.0) are collected on a monthly basis.
- Operational Reliability data (Section 3.0) are collected monthly after acceptance by NASA until responsibility for the software is transitioned to the Navy.
- Delivery Data (Section 4.0) are collected only once. They are collected when NASA accepts the software.

Metrics Data to be Collected	Project Manager Goals and Questions
1.0 Software Characteristics	
1.1 Project CSCI Identification	G1.1: Classify the project in order to uniquely reference it within the database and generate baseline and comparative project management information.
1.1.1—Project name	
1.1.1.1—CSCI name[1]	
1.1.2—Contact person[2]	
1.1.2.1—Contact person's E-mail	
1.1.2.2—Contact person's Organization	
1.1.3—Start date for this CSCI	
1.1.3.1—Number of planned Spiral/Build Iterations	
1.1.4—Estimated final delivery date for this CSCI	
1.1.4.1—Estimated total source code count[3]	
1.1.4.2—Predominant languages used	
1.1.4.3—Estimated percent of functionality provided by COTS and	

[1] When planning multiple releases/deliveries of the same CSCI, treat each delivery separately in Section 2.0 where appropriate based upon development model planned.

[2] Name of person delegated the responsibility for submitting the NASA Software Metrics on a monthly basis for this CSCI.

[3] The count will include software that is to be delivered. The count includes executable statements, and does not include comments, blank lines, or commercial off-the-shelf (COTS) software.

1.1.4.4—Estimated percent of software costs allocated to COTS	
1.1.5—Type of CSCI (flight, ground ops, and 3=any other software development item)	Q1.1: Are there other projects in the database that can be used as a basis or guideline to estimate this project?
1.2 CMM Level	G1.2: Identify developer level of processes maturity, performance, and associated level of oversight needed (i.e., additional resources needed for low-Capability Maturity Model (CMM)-level developers).
1.2.1—Has your organization had a Software Capability Evaluation (SCE) by an Software Engineering Institute (SEI) certified Lead Assessor? [4] 1.2.1.1—If yes, what was the rating?	Q1.2.1: At what CMM Level is the developer?
1.2.2—Estimated CSCI total software cost[5]	Q1.2.2: What is the total software cost?
1.3 First Month Planning Data	
1.3.1—Software accumulated cost for the first month ($K)	
1.3.2—Software cost plan by month through NASA Acceptance	
1.3.3—Total software staff Equivalent People (EP's) for the first month (civil servant and contractor combined)	
1.3.4—Software staffing EP plan by month through NASA Acceptance	
1.3.5—Identification of software products to be completed in the first month	
1.3.6—Identification of all Software Products to be delivered along with their schedule	
1.4 Authorized person for generating reports	
1.4.1—Software manager's name	
1.4.1.1—Software manager's e-mail	
1.4.1.2—Software manager's Center/Company	
2.0 Software Planning, Tracking, and Oversight Data	
2.1 Software CSCI Management Data	G2.1: Determine project progress against plans in order to: —Understand the accuracy of the original estimates. —Determine whether adequate resources are

[4] If any portion of the software development is contracted out, also complete these items for each contract.

[5] Including the cost of development through acceptance, COTS and government off-the-shelf (GOTS) software, middle-ware, and contractors and/or civil servants costs; but not including, however, the cost of maintenance.

	being applied. —Make course corrections (re-planning decisions) during project development to complete on time and within budget. —Know where to apply reserves. —Determine when to look for factors that are impacting plans. —Look for trends to improve future estimating on this project.
2.1.1—Reporting month[6]	
2.1.2—Fiscal year	
2.1.3—Release	
2.1.3.1—Spiral/Build Iteration Number[7]	
2.1.4—Was a new software schedule baseline established during this reporting period?	
2.1.5—Planned Software Accumulated Cost through the subsequent month (see footnote for 1.2.2) 2.1.5.1—Actual Software Accumulated Cost through the recently completed month (see footnote for 1.2.2)	Q2.1.5: What is the difference between planned and actual cost?
2.1.6.1—Planned Completion date for Software Development Plan[8] 2.1.6.1.1—Actual Completion date for Software Development Plan 2.1.6.2—Planned Completion date for Software Requirements Analysis[9] 2.1.6.2.1—Actual Completion date for Software Requirements Analysis 2.1.6.3—Planned Completion date for Software Architectural Design 2.1.6.3.1—Actual Completion date for Software Architectural Design 2.1.6.4—Planned Completion date for Software Detailed Design	Q2.1.6: What is the difference between the planned and actual schedule?

[6] Specify the fiscal month for which this data is being submitted.

[7] If only one is anticipated, specify 1 .If multiple spiral/builds are scheduled, specify the build that this fiscal month's data is being submitted under. If multiple builds are being developed at the same time, then separate reporting sections should be provided for each build

[8] After the initial data is entered, only record the changes from month to month.

[9] Note: the terms used in the following items (i.e. Software Requirements Analysis, Software Architectural Design, Software Detailed Design etc.) are taken from IEEE 12207.0.

[10] Examples may include support of System Integration or System Qualification Testing.

2.1.6.4.1—Actual Completion date for Software Detailed Design 2.1.6.5—Planned Completion date for Software Coding & Testing 2.1.6.5.1—Actual Completion date for Software Coding & Testing 2.1.6.6—Planned Completion date for Software Integration 2.1.6.6.1—Actual Completion date for Software Integration 2.1.6.7—Planned Completion date for Software Qualification Testing 2.1.6.7.1—Actual Completion date for Software Qualification Testing 2.1.6.8—Other Planned Completion date[10] 2.1.6.8.1—Other Actual Completion date	
2.1.7—Total software staff EP's planned for the subsequent month (civil servant and contractor combined) 2.1.7.1—Total software staff EP's actual for the recently completed month (civil servant and contractor combined) 2.1.7.2—Number of unplanned personnel departures in last month	Q2.1.7: What is the difference between planned and actual total staff time?
2.1.8—Identification of planned software products to be completed during the subsequent month[11] 2.1.8.1—Identification of software products completed during the recently completed month	Q2.1.8: What is the actual software production progress against plans?
2.1.9—Current estimate of computer resource utilization at delivery. As appropriate: • Random Access Memory (RAM) • Electronically Erasable Programmable Read Only Memory (EEPROM) • Input/Output (I/O) • Communications Links • Central Processing Unit (CPU)	
2.1.10 – Current estimate of completion data • **Number of planned Spiral/Build Iterations** • **Estimated final delivery date for this CSCI** • **Estimated total source code count**	

[11] Examples of software products are Software Requirements Description, Software Architecture Description, Software Design Description, Source Code Record (e.g. CSC 1, CSC 2, CSC N), Test Plan, Test Procedures, Test Results Report, User Documentation Description, etc.

• **Estimated percent of functionality provided by COTS and estimated percent of software costs allocated to COTS** • **Estimated CSCI total software cost**[5]	
2.2 Software Requirements Management Data	G2.2: Determine software requirements stability in order to: —Determine when resources need to be allocated to solidify requirements. —Determine the growth in requirements. —Determine when to look for factors that are influencing requirements change and/or growth. —Identify when requirements changes are causing cost, schedule, and workforce impacts.
2.2.1 -- Was a new software requirements baseline established during this reporting period?	
2.2.2—Software requirements count[12] at current baseline 2.2.3—Date the current requirements were baselined 2.2.4: Software requirements count at previously entered baseline 2.2.5: Change in total requirements count from previously entered baseline to the current baseline (2.2.1 – 2.2.3)	Q2.2.1: What is the baseline software requirements count? Q2.2.4: How much has the total software requirements count changed since baseline?
2.2.6—Total number of changes to software requirements from the previously entered baseline to the current baseline (Total number of additions, deletions, and modifications.)	Q2.2.5: How many software requirements changes have been accepted from the previously entered baseline to the current baseline?
2.2.7: Percent of change in software requirements (2.2.5/2.2.3 × 100)	Q2.2.6: What is the percent of change in software requirements from the previously entered baseline to the current baseline?
2.2.8—Current requirements count (i.e., end of the month)	Q2.2.7: How much has the total software requirements count changed since baseline?
2.3 Software Testing Data	G2.3: Monitor the number of open and closed Problem Reports (PRs)[13] in order to determine reliability, impacts to schedule, cost, and workforce; and evaluate the likelihood of delivery on time based on the rate at which PRs are being opened and

[12] The software requirements count is the number of unique *shall* statements or other imperatives in the Software Requirements Description for this CSCI. The Automated Requirements Management (ARM) tool is freely available for performing requirements counts at URL http://satc.gsfc.nasa.gov/tools/arm/index.html.

[13] Problem Reports (PRs) as referenced in this text refer to reports for this delivery/release once the software has been delivered for integration testing. Other terms such as *Discrepancy Reports*, or *Failure Reports* are frequently used.

	closed.
2.3.1—Total number open PRs 2.3.2—Total number of closed[14] PRs 2.3.3: Total PRs (2.3.1 + 2.3.4)	Q2.3.1: What is the status of the PRs?
3.0 Operational Reliability	
3.1 Operational Reliability	G3.1: Determine software operational reliability.
3.1.1—Total number of confirmed PRs associated with software functionality [15]	Q3.1.1: What is the total number of confirmed software problems that have been reported since delivery?
4.0 Delivery Data	
4.1 Project Completion Data	G4.1: Determine this project's actuals for use in planning future projects.
4.1.1—Actual final delivery date for this CSCI	Q4.1.1: What is the actual delivery date?
4.1.2—Actual total source code count (see footnote 6)	Q4.1.2: What is the total source code count?
4.1.3: Difference between actual and planned lines of code (4.1.2-1.1.4)	Q4.1.3: What is the difference between actual and planned lines of code at delivery?
4.1.4—Estimated percent of functionality provided by COTS Or 4.1.5—Estimated percent of software costs allocated to COTS	Q4.1.4 What percent of the functions required are being accomplished by COTS?
4.1.6—Date operational	Q4.1.6: What is the date the system went into operation?
5.0 Comments about this project	
5.1—Comments section	G5.1: Provide an explanation concerning any particular metric that was not entered or any item that you feel needs explanation.

[14] A PR is closed once corrective action is successfully implemented and verified.

[15] These PRs are reported after delivery. This data is reported as long as the software is being changed during operations. See IEEE 12207.1, clause 6.10 for details on problem reports.